Agents in multi-user virtual environments

Christian Paul, Ralph Peters and Andreas Graeff

Fraunhofer-Institute for Computer Graphics, Rundeturmstrasse 6, D-64283 Darmstadt, Germany

Abstract
In this paper, we introduce the use of agents in combination with virtual realities as one possibility to enhance future communication systems. These agents support the user and his/her interaction with objects within the virtual world. The main goal of our approach includes the implementation of helpers, personal assistants, and tour guides to make multi-user virtual environments more lively and realistic.

1. Introduction

As an effective and very intuitive vehicle for distributed group interaction, multi-user virtual environments (MVEs) support multimedia object manipulations to enable real-time discussions using video, audio, image or textual information. The graphical user interface (GUI) is the front-end to visualize the distributed virtual 3D space. Multiple users connected from different hosts share a virtual world. Each user can navigate through the world, interact and collaborate with other users, and manipulate objects in the MVE. Video-textured avatars (the graphical user representations) visualize the viewpoint and the behavior of participants. Spatial audio- and video-streams provide a very natural communication between all users.

In a MVE, the system knows which objects are in a scene, how to represent them, and where they move in case they are not static. However, communication among objects, regardless if they are under control of a user or the underlying program, is missing in most cases.

Demonstrations of our MVE at Siggraph97 and ACM97 showed that users have significant difficulties with handling and navigating through the virtual environment. They also requested more liveliness and advanced possibilities to interact with objects in the virtual world.

One possibility to fulfill these demands in virtual realities is the use of software agents. Agents are most relevant and applicable for the use in real-world domains. Their intelligent behavior enables the user to automate and delegate cognitive tasks that were not feasible for machines in the past. Agents support each user individually during a session. They act as a representative of their 'employer' in the task they are assigned to. Several dimensions can classify agent technology: their mobility, whether they are proactive or reactive, their appearance, and their roles[5]. One major issue of agent technology is cooperation among agents. Independent, heterogeneous agents therefore need flexible means to communicate with each other in order to adapt to their environment[3]. Agent Communication Languages (ACL) solve these problems by using communication objects with a specified structure. The most common ACL is the Knowledge Query and Manipulation Language (KQML)[2].

2. Agents in MVEs

As a basis for the work introduced here, we use our multi-user virtual environment (see Figure 1)[1]. Avatars, the graphical representation of participants, can move around in the 3D scenario. In order to improve non-verbal communication, they have video-textured faces and business cards for identification. Walls serve as information displays and may be overlaid by additional video streams or applications. By overlaying, messages and discussion material can be introduced into the scene. The environment allows the user to accomplish a task in an easy and intuitive way although operating in geographically distributed working groups.

A guide agent moves around in the scenario and randomly asks visitors if they request assistance. It has its own representation of the scene and can react upon certain inquiries concerning information about other users or the environment, conferences held in other rooms off the virtual world, and sight-seeing and introductory trips. The guide agent has its own graphical representation just like a regular user. Interaction takes place via buttons mapped on the "face" of the avatar.

Figure 1: *Multi-user virtual environment (MVE)*

For each user an agent is responsible to fulfill access requirements to enter a desired room. It acts on the behalf of its owner. The user is informed about all possibilities of payment so that s/he just has to make the final decision whether or not the next room should be entered given these premises. A door agent, which asks for a certain access code to a room, communicates with the user agent[4].

The next step will be the realization of object behavior. The central idea is that every object will provide its own functionality. This will be realized by creating object-agent pairs. Every agent is responsible to supply necessary functions to

- create new objects,
- administrate the object (event-handling),
- allow modification,
- control the object (representation, position, general behavior), and
- erase objects.

This will offer the possibility to enhance usability and realism of virtual environments by combining 3D objects with intelligent agents, and to explore new aspects evolving from this combination.

3. Architecture

The architecture is split in two parts: the multicast net and the agent net. The multicast net consists of a high-speed multicast backbone (MBone). It connects the MVE Clients and is the basis for transmitting continuous audio and video feeds as well as all changes that occur in the virtual environments. The agent net interconnects all agents. This partition into two independent nets results from the different requirements regarding communication and scalability. While agents rely on a secure submission of their messages, video and audio feeds do not have such requirements; they need high-speed connections to allow real-time transmissions. Agents come to full effect when they form agent societies, consisting of a large number of agents acting in and therefore supporting a relatively small number of virtual environments. So-called MVEAgents link the two subnets together. They accept messages from either side and enable the agents to control objects in the environments, as well as transferring user inputs from the MVE Clients to the corresponding agents. Note that agents may or may not reside on the same machine as the MVE Clients.

To build and execute agents we use “A Simple Agent Platform” (ASAP)[4]. This platform is a framework and runtime environment that helps to develop and execute new agents in an easy and uncomplicated way. ASAP is based on Java and KQML (Knowledge Query and Manipulation Language[2]) and provides agent templates that enable the programmer to develop software agents rapidly. Agents and the underlying agent communication platform ASAP can easily be incorporated in existing systems due to their versatile structure and platform independence.

4. Conclusion

In this article we proposed the integration of intelligent agents and multi-user virtual environments to enhance the usability of 3D user interfaces. At the present time, stable prototypes of the MVE system and the agent platform ASAP exist. By integrating both of these research projects, this work creates a framework to extend objects in MVEs with high level behaviors. Using the resulting system, different agent behaviors and their usability can be evaluated. Experiments with already implemented agents showed a significant simplification of the tasks and increased usability for the user.

References

1. Encarnacao, J.L., Noll, S., Macedonia, M.R., Goettsch, A., Schiffner, N., Graeff, A., “TRADE: A Transatlantic Research and Development Environment”, *Proceedings of the 1997 International Conference on Virtual Systems and MultiMedia*, IEEE Publication, Geneva (September 1997)

2. Finin, T., Fitzson, R., McKay, D., McEntire. R., “KQML as an Agent Communication Language”, ACM Press (November 1994)

3. Nwana, H.S., “Software Agents: An Overview”, Cambridge University Press, Cambridge, UK (1996)

4. Peters, R., Graeff, A., Paul, C., “Integrating Agents into Virtual Worlds”, *Proceedings of the International SIGGRAPH Workshop on New paradigms in Information Visualization and Manipulation* Las Vegas, Nevada, USA (November 1997)

5. Wooldridge, M.J., Jennings, N.R. “Intelligent Agents: Theory and Practice”, *The Knowledge Engineering Review* **10** (2) (1995)